MVS and FACOM Installation

The installation steps described in this document must be followed in sequence in order to install Com-plete successfully. You are recommended to use the provided examples up to successful initialization in order to ensure that the installation was successful. Once a functional system is available, you can start customizing the product as required.

The installation tape is created using IBM standard labels with a volume serial number of COM*vrs*. The tape can be ordered in any format supported by the SOFTWARE AG tape system.

Notes:

1. The notation *vrs* in the table below stands for version, release and system maintenance level.
2. For a description of installing the ADASVC, refer to the Adabas documentation.
3. The Com-plete Batch utilities require access batch (that is, the Adabas SVC must be installed).

This document covers the following topics:

The Installation Tape

Tape Contents

Note:
The files may not be on the installation tape in the order shown below: please consult the report of your tape creation system.

File Nr.	File Name	Description
1	SMT111.TABS	SMA tables
2	COMvrs.SRCE	Source data set
3	COMvrs.SRCT	TPF source data set
4	COMvrs.LOAD	Load library
5	COMvrs.LOA3	TPF load library
6	COMvrs.BASE	System file
7	COMvrs.USER	User ID file
8	COMvrs.CTLG	Catalog file
9	COMvrs.NS22	NSPOOL INPL data set

File Nr.	File Name	Description
10	WAL532.LOAD	Adabas limited load library
11	WAL532.SRCE	Adabas limited source library
12	COMvrs.CBLK	Com-plete control block listings
13	COMvrs.JOBS	Com-plete sample JCL library
14	APSvrs.SARG	Not currently used with Com-plete
15	APSvrs.SXML	Not currently used with Com-plete
16	APSvrs.LB00	Not currently used with Com-plete
17	APSvrs.LD00	SMARTS load library
18	APSvrs.LE00	Not currently used with Com-plete
19	APSvrs.SRCE	SMARTS source library
20	APSvrs.LDnn	SMARTS load library update (current patch level)
21	APSvrs.SRnn	SMARTS source library update (current patch level)
22	HTPvrs.LOAD	HTTP server load library
23	HTPvrs.INPL	HTTP server INPL data set
24	HTPvrs.UPDW	Natural Web interface update INPL dataset
25	HTPvrs.GIFS	PDS containing GIF images for Natural Web Interface demo
26	HTPvrs.Lnnn	Most current patch level of HTPvrs.LOAD
27	HTPvrs.Snnn	Most current patch level of HTPvrs.SRCE
28	HTPvrs.Innn	Most current patch level of HTPvrs.INPL
29	HTPvrs.JOBS	HTTP server sample JCL library

Copying Contents of the Tape to Disk

Step 1: Unloading the Tape

1. Copy all files on the installation tape to disc using the IEBCOPY utility for the source and load data sets, and the IEBGENER utility for the system data. Please refer to *Sample JCL for MVS and FACOM* for sample JCL.

   Note:
   APSvrs.LDnn is mandatory for Com-plete 6.2

2. Check for a README file on the delivered source. If a README file exists, it may contain important information concerning the installation and migration procedure.

Installation Steps

Step 1: Authorize Com-plete

Note:
Com-plete must run as an authorized user. To achieve this, you need only place module TLIN*xx* in an authorized data set, where *xx* can stand for "OS" for MVS systems and "F4" for FACOM systems. Use one of the following alternative steps.

1. Copy the module from the APS*vrs*.LDnn to a library that is already authorized and use the library as the STEPLIB for Com-plete. Sample JCL to copy the module is contained in member JCLINST1. A sample link job for the module is provided in JCLINST9. If you are a FACOM installation, you must use this, as the COPYMOD is not supported.

   Note:
   If you do not use JCLINST1, you must ensure that whatever method you use to copy the module retains the correct AMODE/RMODE attributes and authorization code for the module.

   Or:
   Authorize the distributed Com-plete load library permanently by adding an entry for the data set in member IEAAPF*nn* of library SYS1.PARMLIB, where *nn* is the suffix used at your site.

   Note:
   This requires an IPL of the system. You can avoid having to perform an IPL for the initial installation if you have a product installed that allows automatic authorization of the data set.

Step 2: Allocate and Initialize the Com-plete Non-VSAM Data Sets

- Allocate the data sets needed to run Com-plete by modifying sample job JCLINST2 to reflect appropriate size and placements of data sets.

  This job allocates all non-VSAM data sets:

  ```
  COM.USER.SOURCE
  COM.USER.LOAD
  COM.MAPS
  ```

Step 3: Allocate and Initialize the Com-plete VSAM System Data Containers

- Only for a new installation or for an upgrade from a version *lower* than 4.6.:

  Allocate, initialize and load the Com-plete system information data sets by modifying sample job JCLINST3 to reflect appropriate size and placement of the files. Please refer to the description of using VSAM in the Com-plete System Programming documentation for more information on this VSAM file.

Step 4: Allocate and Initialize the Capture Data Set(s)

- Required only if capture is used at your installation.

  Com-plete capture data sets are defined as VSAM data sets in the current version. Sample job JCLINST4 provides JCL to allocate and initialize two capture data sets. You must modify the job to reflect appropriate size and placement of the capture data sets.

Step 5: Allocating and Initializing the Com-plete SD Dataset

- Required.

  The Com-plete SD dataset must be defined as a VSAM dataset in the current version. For detailed information on this file, please refer to the description of using VSAM in the Com-plete System Programming documentation. To allocate and initialize the SD dataset, modify sample job JCLINSTE to reflect your requirements and run this job.

Step 6: Allocate and Initialize the Com-plete Spool Data Set

- Required.

  The current version of Com-plete requires the spool data set to reside on a VSAM file. To allocate and initialize this file, modify sample job JCLINST5 to reflect appropriate size and placement of the data set. Then run the job.

  Note:
  The DD name used for this dataset in the Com-plete startup procedure must be *COMSPL*.

Step 7: Install the Com-plete Batch Interface

Required.

The Com-plete BATCH utilities and migration jobs require ACCESS when performing some or all of their functions.

Batch applications that use Com-plete services communicate with the target Com-plete via ACCESS. This communication is based on an Adabas SVC (ACCESS SVC) and a DBID (ACCESS NODE ID).

The Com-plete BATCH interface module loads a module with name ACSTAB and search this module for an entry name BATCH.

1. Modify the sample ACSTAB module on the source data set to reflect the system requirements for SVC and NODE ID.

2. Assemble and linkedit the module into the COM.USER.LOAD library. Modify sample job JCLINSTF to do this. The library specified as SYSLMOD must then be contained in the STEPLIB DD chain of the batch jobs.

   Note that the DBID chosen may be greater than 255 and that the value of both DBID and SVC must correspond to the values of ACCESS-ID and ACCESS-SVC specified in the SYSPARM member.

Step 8: Assemble and Link the Adabas Link Module

- This module is delivered in source format as ADALCO on the Adabas distribution libraries. It must be assembled and linked in to a dataset contained in the COMPLIB concatenation.

If the version of Adabas you use is less than 6.2, the following source change must be applied to module ADALCO before assembly:

Locate the line

```
BP GOTWAIT
```

and replace it by

```
BNZ GOTWAIT
```

Note:
This module *must* be linked with AMODE=31

Step 9: Eligible Device Table Considerations

Required only if you are installing Com-plete under versions of MVS prior to ESA V4.

The COM.SOURCE data set contains the operating system-dependent module UDSEDT. This module interrogates the EDT (Eligible Device Table) for UDS. The supplied module runs for all levels of MVS/ESA after V4. If you are installing Com-plete under an MVS version prior to ESA V4, proceed as follows:

1. Simply assemble the UDSEDT module (the module itself determines the level of MVS for which it is being assembled).
2. Relink UDS using the linkedit step within the applicable operating system link job. See the description of Modifications to Com-plete Modules in the Com-plete System Programming documentation for more information on Com-plete linkedits.

   See the System Programming documentation for more details on updating Com-plete utilities.

Step 10: Install the Com-plete High Level Language Interface

- Optional.

  Applications that use Com-plete functions must be linked with the Com-plete high level language interface (HLLI) modules. These modules provide a standard interface between the application and Com-plete.

  These modules are provided on the supplied load data set. If you are migrating from a previous version of Com-plete, some applications may need to be relinked with the new version of the HLLI modules. For details, refer to the description of Migration procedures.

Step 11: Install the JES Server Interface

The JES Interface modules are distributed in source to facilitate support of different levels of JES. DO NOT modify the code else Software AG cannot provide support for the modules.

The highest JES level supported is indicated in the source member. If the JES to which you wish to interface is at a higher level it might work, however it is recommended to contact the local support center for any updates.

The JES Interface uses the Com-plete sysparm SERVER to initialize, terminate and communicate with the JES interface server. Prior to defining any Com-plete sysparms, you must follow one of the following procedures.

1. **Installing the Common JES2/JES3 Server Interface**

   This interface requires the following prerequisites:

OS/390 Version 2.6 or above	
JES2	The PTFs for APARs OW35104, OW36019, OW44349 and OW52411 must be applied.
JES3	The PTFs for APARs OW34753, OW36022 and OW44298 must be applied. See also Information APAR II11784 for recommended JES3 SAPI maintenance.

   1. For JES2, assemble TTJ2MVS and TTJIMVS using your current JES2 MACLIB. As an example you can use JCLINSTC.

      For JES3, assemble TTJ3MVS and TTJIMVS using your current JES3 MACLIB. As an example you can use JCLINST8.

      Note:
      There is no need to assemble the common interface module JESCSERV, since it is neither OS nor JES version dependent.

   2. Add the following SERVER Statement to the Com-plete sysparms:

      For JES2:

      ```
      SERVER=(ssss,TLINJES2)
      ```

      For JES3:

      ```
      SERVER=(ssss,TLINJES3)
      ```

      where:

ssss	is a unique server name within your Com-plete.

2. **Installing the FACOM JES Interface**

   In previous versions, the JES interface required the JES checkpoint dataset (default name SYS1.SYSCKPT) to be cataloged. In Com-plete 5.1.3 and above, the name and volume of this dataset can be specified as subparameters of the SERVER SYSPARM used to define the JES server in Com-plete. If this information is not provided, the Com-plete JES interface obtains it from the JES address space at initialization time.

- Specify the SERVER SYSPARM as follows:

```
SERVER=(ssss,TLINJES2,mmmmmmmm,subsys,ckptvol,dsnprfx)
```

where:

ssss	is a unique server name within Com-plete (e.g., JES).
mmmmmmmm	is the name of the JES server module. You are recommended to specify this parameter only if you have renamed the JES server module for some reason. If the parameter is not specified, the TLINJES2 module selects the default module based on the level of MSP.
subsys	is the name of the JES subsystem. This parameter need only be specified if you have specifically renamed the JES subsystem. If this parameter is not specified, TLINJES2 will use the standard name.
ckptvol	is the volume name of the JES checkpoint dataset as specified on the &CHKPT parameter of JES. This parameter is required if your machine (CPU) does not support the S/390 architecture. If this parameter is not specified, the JES interface module will obtain it from the JES address space.
dsnprfx	is the dataset name prefix of the JES system dataset (CKPT and SPOOL) as specified on the &DSNPRFX parameter of JES. If this parameter is not specified, the JES interface module will use SSYS1 by default.

For example, the following SERVER sysparm will enable the Com-plete FACOM JES server to run under non-S/390 archtictures (e.g., M780):

```
SERVER=(JES,TLINJES2,,,VOL001)
```

Note:
If an error occurs during startup, a return and reason code is written in the message to the console. The reason codes can be found in member TTJIEQU on the supplied source dataset.

3. If your IBM OS version does not meet the prerequisites to use the common JES interface, you can still use the old JES interfaces JES2SERV and JES3SERV. They are distributed in source to facilitate the support of different levels of JES. DO NOT modify the code, or Software AG cannot provide support for the modules anymore.

   Installing the JES2 Server Interface

   1. Assemble the JES2 Server interface module JES2SERV.

      The JES2SERV module found on the source library must be assembled using the JES2 SYS1.SHASMAC (OS/390) or SYS1.HASPSRC (MVS) data set for the JES2 with which you wish to work and the SYS1.MACLIB and SYS1.AMODGEN (SYS1.MODGEN for ESA V4) under which the JES2 subsystem will work. Member JCLINSTB on this source data set is a sample JCL stream to do this. You can give the load module resulting from this assembly and link any name you like.

   2. Assemble the MVS Active display format module TTJIMVS.

      The TTJIMVS module found on the source library must be assembled using the SYS1.MACLIB and SYS1.AMODGEN (SYS1.MODGEN for ESA V4) for the MVS level on which the JES2 interface will work. Member JCLINSTC on this source data set is a sample JCL stream to do this. The load module resulting from this assembly and link must be called TTJIMVS.

      Note:
      The two modules created by the jobs as described in steps 1 and 2 above must be linkedited with RMODE=ANY and AMODE=31, otherwise abends may occur at initialization.

   3. Specify the SERVER sysparm as follows:

      ```
      SERVER=(ssss,TLINJES2,mmmmmmmm,subs)
      ```

      where:

ssss	is a unique server name within your Com-plete (e.g., 'JES2').
mmmmmmmm	is the name of the assembled and linked JES2 module.
subs	is the name of the JES2 subsystem on your system. The default is "JES2". Use this parameter if your JES2 system has a different name, or if you have more than one JES2 subsystems installed.

      For example, the statement

      ```
      SERVER=(JES,TLINJES2,JES2SERV,JES2)
      ```

will cause Com-plete to use the JES2 server module JES2SERV, which will in turn attempt to interface with JES2 subsystem JES2, while the Com-plete server name will be JES.

Note:
If an error occurs during startup, a return and reason code is written in the message to the console. The reason codes can be found in member TTJIEQU on the supplied source dataset.

Installing the JES3 Server Interface

1. Allocate the JES3 communication file.

   Note:
   Applies only when installing Com-plete for the first time.

   The installation job JCLINST7 allocates and initializes the MVS file used by the DSP and the JIM. After this installation job has successfully completed, add the following DD card to the Com-plete start up procedure.

   ```
   //UQJ3JOBS  DD  DSN=COM.JES3.UQJOBS,DISP=SHR
   ```

   and add the following DYNALLOC statement to the JES3 initialization deck.

   ```
   DYNALLOC,DDNAME=UQJOBS,DSNAME=COM.JES3.UQJOBS
   ```

2. Assemble the JES3 interface modules JES3SERV and IATUQJ3.

   The JES3 modules JES3SERV and IATUQJ3 found on the source library must be assembled using the JES3 SYS1.JES3MAC data set for the level of JES3 with which you work and the SYS1.MACLIB and SYS1.AMODGEN (SYS1.MODGEN for ESA V4) under which the JES3 subsystem will work.

   Member JCLINST8 is a sample JCL stream to do this.

   Note:
   Both JES3SERV and IATUQJ3 contain assembly variables which must be set to correspond to the version of JES3 under which they are to run. Module IATUQJ3 is upward compatible with previous releases of Com-plete (and also Natural system server).

   The assembled and linked JES3SERV module must be placed in a load library contained in the COMPINIT concatenation. The module IATUQJ3 must be placed in a load library contained in the JES3 startup concatenation.

3. Add the DSP dictionary entry.

   Note:
   Applies only when installing Com-plete for the first time.

   Add a DSP dictionary entry for the UQJ3 DSP, this update applies to the JES3 module IATGRPT.

   You must first verify that the USERMOD name (SAG0001) and the sequence numbers do not conflict with existing installation modifications to JES3.

The following SMP statements are a sample to install the UQJ3 DSP entry for XA22 (JES3/HJS2220).

```
//SMPPTFIN  DD  *
++ USERMOD (SAG0001)
++ VER (Z038) FMID(HJS2220)
++ SRCUPD (IATGRPT)
./ CHANGE NAME=IATGRPT
         TITLE 'DSP FOR UQJ3'
SAG001
*-------------------------------------------------------------
*SAG001          DSP DIRECTORY ENTRY FOR UQJ3 DSP
                 SAG001
*-------------------------------------------------------------
*SAG001
UQJ3     IATYDSD PRTY=10,XABLE=YES,DRVR=IATUQJ3
SAG001
         SPACE
SAG001
//SMPCNTL DD   *
 SET BDY(GLOBAL) .
 RECIEVE S(SAG0001) SYSMOD .
 SET BDY(XATRGZN) .
  APPLY S(SAG0001) .
/*
```

Once the above changes have been made, restart JES3 to pick them up. A HOTSTART restart can be done and the Com-plete DSP can then be started by entering "*X UQJ3", the DSP will continue to run and will restart itself over HOTSTART and WARMSTARTs.

Note:
The DSP must be active for the first time before you start Com-plete.

4. Specify the SERVER sysparm as follows:

   SERVER=(ssss,TLINJES3,mmmmmmmm)

   where

ssss	is a unique server name within your Com-plete (e.g. JES3)
mmmmmmmm	is the name of the assembled and linked JES3 module (default is JES3SERV)

Step 12: Install the UDEBUG Environment

Optional.

Com-plete provides an online debugging utility (UDEBUG). In order to establish the working environment, proceed as follows:

1. Allocate two UDEBUG profile and symbol datasets as described in Com-plete Files and Associated User Files in the System Programming documenation.

2. Include the list of required UDEBUG residentpage modules in your SYSPARM member. A list of these modules is given in member DBUGSAMP in the supplied source datasets.

   The UDEBUG facility is described in detail in the Com-plete Utilities documentation.

Step 13: Enable VTAM Interface

- To enable the VTAM interface, an ACB and Logmodes must be defined in the VTAM libraries. Refer to the description of the VTAM interface in the Com-plete System Programming documentation for the required definitions and parameters.

Step 14: Check Com-plete Startup Parameters (SYSPARM)

- A sample SYSPARM member is distributed in the Com-plete source library. This member can be used as a base for Com-plete customization. Although most parameters need not be changed, ensure that the following parameters reflect your environment:

Parameter	Value
ACCESS-SVC	Specify the correct Adabas SVC number (see Step 7).
ACCESS-ID	Specify the database ID for ACCESS (see Step 7).
PATCHAR	Choose a unique single character for this installation (see Com-plete Startup Procedure in the System Programming documentation).
SECSYS	Specify the security system running at your site (see Software Interfaces in the Com-plete System Programming documentation for details on the SECSYS parameter).
SERVER *	Required for Com-plete 6.2 and above: `SERVER=(OPERATOR,TLINOPER)SERVER=(COMPLETE,TLINCOMP)` Optional: Code 1 SERVER statement for each installed server according to the required syntax: JES SERVER — See Step 11 above. CONSOLE SERVER — See Extended Console Server in the System Programming documentation. Natural Buffer Pool Manager — See Natural Buffer Pool Manager in the System Programming documentation. These SERVER statements must be put after the Com-plete server.
VTAMAPPL	Must specify Com-plete’s ACB name (see Step 13).

* You can optionally code one SERVER statement for each installed server according to the required syntax:

Step 15: Install NSPOOL

Optional.

The printout spooling facility NSPOOL is described in the Com-plete Application Programming documentation. The required dataset for running NSPOOL is on the Com-plete installation tape and is loaded as part of the Com-plete installation. The dataset name is COM*vrs*.NS22.

1. Run an INPL of NSPOOL.

2. If NSPOOL is to be used in a Natural environment running under Natural Security, you must stow the following modules in the Natural Security system library; otherwise, proceed to the next point.

   ```
   SPSE01-N,
   SPSE02-S,
   SPSE01-&,
   SPSE02-&,
   SPSE03-&
   SPAPPL-N *
   SPUSER-N *
   ```

   * required only for Natural Security version 2.2.8 and above.

   See also the section **Default Authorization** below.

3. With Natural 2.2, you must place the following modules in the Natural system library and delete them from the NSPOOL library:

   ```
   SPSEC01N
   SECERROR *
   NSCDAU   *
   NSCDU    *
   NSCLU-G *
   NSCLU-M *
   SNAAREXT *
   SNAASEXT *
   SNAUREXT *
   SNAUSEXT *
   ```

   * not required for Natural Security version 2.2.8 and above

4. Set the parameter MEMRES in the NATPARM module to 10k to allow Natural to load the program at execution time.

5. The Natural startup application (for example, NAT22) must be cataloged with ULIB as PV (privileged).

Step 16: Install the POSIX Server

The Telnet tn3270 support, HTTP server and LPD print server support of Com-plete 6.2 require the POSIX runtime to be active.

All load modules used by the POSIX runtime are contained in APS*vrs*.LDnn which is mandatory for Com-plete 6.2 even if you do not plan to use any POSIX-based components.

1. To activate the POSIX runtime, add the following SERVER statement to your SYSPARMs:

   ```
   SERVER=(POSIX,PAENKERN)
   ```

   Note:
   Servers are started in the order in which their SERVER= statements are found in the SYPARMs. The servers OPERATOR, POSIX, COMPLETE must be defined in this order.

2. POSIX runtime parameters can be specified directly in the Com-plete SYSPARM member, or in a separate member concatenated to the Com-plete SYSPARMs in the startup JCL procedure. The POSIX runtime may require additional parameters to be set, please refer to the SMARTS Installation and Operations for a detailed description of those parameters. In particular, the above mentioned TCP/IP based Com-plete components need to know which TCP/IP stack to use. This definition must be made using the following POSIX runtime parameter:

   ```
   CDI_DRIVER=('tcpip,PAAxSOCK [,...]')
   ```

   The HTTP server requires an additional CDI driver definition:

   ```
   CDI_DRIVER=('cgistdio,HAANPCGI')
   ```

   Note:
   Servers are started in the order in which their SERVER= statements are found in the SYPARMs. The servers OPERATOR, POSIX, COMPLETE must be defined in this order.

Step 17: Activate the TELNET Server

Prerequisite: the POSIX server must be installed.

- The TELNET server is activated by simply assigning an IP port number for it. Contact the IP administrator of your host for a port number, and specify it using the Com-plete startup parameter TELNETPORT=port_number.

  No SERVER= statement is required for the TELNET server.

Step 18: Install the HTTP Server

Install the HTTP server as described in the HTTP Server documentation.

In general, the following steps are necessary:

1. Make sure the POSIX runtime is setup properly, including the CDI drivers (see above).

2. Add a SERVER statement to your SYPARMs:

   ```
   SERVER=(HTTP,HAENSERV,CONF=HAANCONF).
   ```

   Note that this server definition must be placed behind the definitions of the servers OPERATOR, POSIX, and COMPLETE in the SYSPARMs.

3. Use the sample module HAANCONF from the HTTP server source library as a pattern to setup your own configuration. This is the place where you specify, for example, the port number your HTTP server will be using. Assemble this module.

Startup Procedure

Com-plete can be run as a started task or as a job. In order to start Com-plete, proceed as follows:

Step 1: Modify Procedure COMJCL

- Review sample procedure COMJCL in the source library for correct data set names. Then either copy the procedure to a procedure library to be started, or use it as a basis for a job to be submitted. If you are a FACOM installation, you must change the startup program name as described in the COMJCL procedure.

  Note:
  For more information on the data sets and their purpose, see the Com-plete System Programming documentation.

Step 2: Initialize Com-plete for the First Time

- Start Com-plete either by submitting the created job or starting the created procedure. Monitor the startup messages carefully for any signs of problems. When the initialization is complete, a message to that effect is written to the operator console.

  A message is also sent to the console to notify you that the VTAM and ACCESS interfaces have initialized.

Step 3: Log On

- Log on through VTAM to the specified VTAM application name. The Com-plete ULOG menu is displayed, prompting you for user ID and password. Logon to Com-plete using user ID SAGADMIN and password ADMIN.

  This user ID has a definition on the Com-plete system data container with the required authorization to define more user IDs to the system using the user ID maintenance facility (see the Com-plete Utilities documentation).

  After defining other user IDs, you are recommended to change the password for user SAGADMIN and use this user ID in emergency cases only.

What Next?

Com-plete is now up and ready for work. How you continue depends on whether you have installed Com-plete for the first time or whether you are migrating from a previous version.

- If you have installed Com-plete for the first time, no further migration is necessary. You can continue with customization steps described in the System Programming documentation and the Utilities documentation.

- If you are migrating from a previous version of Com-plete, see **Migration**.